DETAILED ACTION

1. Claims 1-23 have been considered. Claims 3-5, 8-9, 11-17 have been amended as per Applicants' request. Claims 18-20 have been canceled as per Applicants' request. New claims 21-23 have been added as per Applicants' request.

Papers Submitted

2. It is hereby acknowledged that the following papers have been received and placed of record in the file: Specification, Abstract, Claims, Drawings, Documents submitted with 371 Applications, Certified Copy of Foreign Priority Application, Preliminary Amendment as filed 03 January 2006; IDS as filed 03 January 2006; Oath or Declaration, IDS as filed 21 April 2008; IDS as filed 04 September 2008; Request for status of Application as filed 03 September 2010.

Priority

3. Receipt is acknowledged of papers submitted under 35 U.S.C. 119(a)-(d), which papers have been placed of record in the file.

Information Disclosure Statement

4. The information disclosure statements filed 03 January 2006 and 21 April 2008 fails to comply with 37 CFR 1.98(a)(2), which requires a legible copy of each cited foreign patent document; each non-patent literature publication or that portion which caused it to be listed; and all other information or that portion which caused it to be listed. It has been placed in the application file, but the information referred to therein has not been considered. There were no copies of the European Patents cited and the Examiner could not determine which NPL filed was correct, since the NPL was either missing titles or the titles apparent did not match the NPL titles on the PTO-1449.

5. The information disclosure statement (IDS) submitted on 04 September 2008 was filed is in compliance with the provisions of 37 CFR 1.97. Accordingly, the information disclosure statement is being considered by the examiner.

Allowable Subject Matter

6. Claim 7 is objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

7. The following is a statement of reasons for the indication of allowable subject matter: The prior art searched and found have not taught the combination of limitations found in the claims. More specifically, the prior art searched and found have not taught, in combination with other limitations in the claim, in the event of interruption of the object creation instruction, incomplete initialized objects are created and clearly differentiating between incompletely initialized objects and completely initialized objects.

Double Patenting

8. Claim 21 is objected to under 37 CFR 1.75 as being a substantial duplicate of claim1. When two claims in an application are duplicates or else are so close in content that they both cover the same thing, despite a slight difference in wording, it is proper after allowing one claim to object to the other as being a substantial duplicate of the allowed claim. See MPEP § 706.03(k).

Claim Rejections - 35 USC § 103

9. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

10. Claims 1-6, 8-21, and 23 are rejected under 35 U.S.C. 103(a) as being unpatentable over Breid et al., U.S. Patent Number 5,860,092 (herein referred to as Breid) in view of Free On-Line Dictionary of Computing ©1995-2003 (herein referred to as FOLDOC).

11. Referring to claims 1, 21, and 23, taking claim 1 as exemplary, Breid has taught a processor architecture in which the access to a memory occurs via pointers which refer to objects, characterized in that

pointers are stored in a pointer area and data is stored in a data area separately from one another in the objects (Breid Abstract; column 1, lines 24-56; Figure 1; column 2, line 41 to column 3, line 37),

the pointers containing a memory address of the object to which they refer and the objects being provided with attributes which are stored in the object itself and which describe a length of the pointer area and a length of the data area (Breid Abstract; column 1, lines 24-56; Figure 1; column 2, line 41 to column 3, line 37), and

the pointers are used for the access to objects in the memory (Breid Abstract; column 1, lines 24-56; Figure 1; column 2, line 41 to column 3, line 37).

12. Breid has not taught a register set with separate data and pointer registers. However, Breid has taught separate data and pointer memories (Breid Abstract; column 1, lines 24-56; Figure 1; column 2, line 41 to column 3, line 37). FOLDOC has taught using registers (FOLDOC "register set", "register" ©2000). A person of ordinary skill in the art at the time the invention was made, and as taught by FOLDOC, would have recognized that registers are typically faster than other memory.

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the registers of FOLDOC in the device of Breid to increase speed.

13. Referring to claim 2, Breid in view of FOLDOC has taught the processor architecture according to Claim 1, characterized in that the processor ensures that every pointer contains only either a predefined null value (FOLDOC "null" ©2003) or the memory address of an existing object (Breid Abstract; column 1, lines 24-56; Figure 1; column 2, line 41 to column 3, line 37).

14. Referring to claim 3, Breid in view of FOLDOC has taught the processor architecture according to Claim 1, characterized in that a instruction set having separate instructions for data and pointer operations is used (Breid Abstract; column 1, lines 24-56; Figure 1; column 2, line 41 to column 3, line 37).

15. Referring to claim 4, Breid in view of FOLDOC has taught the processor architecture according to Claim 1, characterized in that load and store operations for pointers exclusively load pointers from the pointer areas of the objects into the pointer registers and/or store the content of pointer registers into the pointer areas of the objects, and load and store operations for data exclusively load data from the data areas of the objects in data registers and/or store the content of data registers in the data areas of the object (Breid Abstract; column 1, lines 24-56; Figure 1; column 2, line 41 to column 3, line 37).

16. Referring to claim 5, Breid in view of FOLDOC has taught the processor architecture according to Claim 1, characterized in that a instruction set having an object creation instruction is used, which initializes all pointers in the pointer area of a created object with a null value before the created object may be accessed (FOLDOC "null" ©2003).

17. Referring to claim 6, Breid in view of FOLDOC has taught the processor architecture according to Claim 5, characterized in that the object creation instruction may be interrupted and resumed at a later point in time (FOLDOC "interrupt" ©1995).

18. Referring to claim 8, Breid in view of FOLDOC has taught the processor architecture according to Claim 1, characterized in that the processor supports constant objects which are kept in a separate memory area that is exclusively read at program runtime, and pointers to constant objects are clearly identified by the processor (Breid Abstract; column 1, lines 24-56; Figure 1; column 2, line 41 to column 3, line 37).

19. Referring to claim 9, Breid in view of FOLDOC has taught the processor architecture according to Claim 1, characterized in that a program stack is used, which is divided into a pointer stack area and a data stack area, a length of the occupied part in each of the two stack areas being indicated by a stack index, which is managed in a data register reserved for this purpose (Breid Abstract; column 1, lines 24-56; Figure 1; column 2, line 41 to column 3, line 37) (FOLDOC "stack" ©1995).

20. Referring to claim 10, Breid in view of FOLDOC has taught the processor architecture according to Claim 9, characterized in that a instruction is used for pushing a pointer onto the pointer stack, which both stores the corresponding pointer onto the pointer stack and also increases the pointer stack index in a non-interruptible way (FOLDOC "stack" ©1995).

21. Referring to claim 11, Breid in view of FOLDOC has taught the processor architecture according to Claim 1, characterized in that the processor supports static objects which are kept in a separate memory area, which is managed by an operating system, and pointers to static objects

are clearly identified by the processor (Breid Abstract; column 1, lines 24-56; Figure 1; column 2, line 41 to column 3, line 37).

22. Referring to claim 12, Breid in view of FOLDOC has taught the processor architecture according to Claim 11, characterized in that static objects are used for the program stack and that the attributes contained in the object describe the length of an actually occupied part of the stack area in the event of an inactive program stack (Breid Abstract; column 1, lines 24-56; Figure 1; column 2, line 41 to column 3, line 37).

23. Referring to claim 13, Breid in view of FOLDOC has taught the processor architecture according to Claim 1, characterized in that an attribute register is assigned to every pointer register, in which the attributes of the object to which the pointer in the pointer register refers are written (Breid Abstract; column 1, lines 24-56; Figure 1; column 2, line 41 to column 3, line 37).

24. Referring to claim 14, Breid in view of FOLDOC has taught the processor architecture according to Claim 1, characterized in that a pipeline having an additional pipeline stage is used for loading the attributes (Breid Abstract; column 1, lines 24-56; Figure 1; column 2, line 41 to column 3, line 37).

25. Referring to claim 15, Breid in view of FOLDOC has taught the processor architecture according to Claim 1, characterized in that an attribute cache is used (Breid Abstract; column 1, lines 24-56; Figure 1; column 2, line 41 to column 3, line 37).

26. Referring to claim 16, Breid in view of FOLDOC has taught the processor architecture according to Claim 1, characterized in that an RISC instruction set is used (Breid Abstract; column 1, lines 24-56; Figure 1; column 2, line 41 to column 3, line 37).

27. Referring to claim 17, Breid in view of FOLDOC has taught the processor architecture according to Claim 1, characterized in that the processor performs automatic garbage collection (FOLDOC "garbage collection" ©1997).

28. Referring to claim 22, Breid in view of FOLDOC has taught a device comprising:

a main processor having a processor architecture in which access to a memory occurs via pointers which refer to objects, wherein the pointers are stored in a pointer area and data is stored in a data area separately from one another in the objects (Breid Abstract; column 1, lines 24-56; Figure 1; column 2, line 41 to column 3, line 37), the pointers containing a memory address of the object to which they refer and the objects being provided with attributes which are stored in the object itself and which describe a length of the pointer area and a length of the data area (Breid Abstract; column 1, lines 24-56; Figure 1; column 2, line 41 to column 3, line 37),

the main processor further including a register set with separate data and pointer registers (FOLDOC "register set", "register" ©2000), of which the pointer registers are used for the access to objects in the memory (Breid Abstract; column 1, lines 24-56; Figure 1; column 2, line 41 to column 3, line 37); and

a coprocessor operative to perform automatic garbage collection and being closely coupled to the main processor for efficient synchronization (FOLDOC "coprocessor" ©1995).

Conclusion

29. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

a. Sumita, U.S. Patent Application Publication 2003/0196038 and U.S. Patent Numbers 6,574,711; 7,260,683; 7,444,475, have taught a cache tag memory separate for cache data memory and accessing data via the tag cache.

b. Holden, U.S. Patent Application Publication 2004/0184470, has taught a queue pointer stored separately from data containing an address pointer to the buffer and length of the data, and data in the buffer holding control information and the data.

c. Vogt et al., U.S. Patent Number 5,897,656, has taught variable sized tag memory and the sizes corresponding to a page address and offset address of a cache line.

d. Yoshimura, U.S. Patent Number 5,907,716, has taught data sets having different length and a separate pointer buffer for storing write pointer values.

e. Clarke et al., U.S. Patent Number 6,751,583, has taught initializing tables with null pointers.

f. Lueh, U.S. Patent Number 6,895,579, has taught throwing exceptions while initializing registers.

g. Ware et al., U.S. Patent Number 7,454,555, has taught tag memory separate from memory and address memory based upon the tags.

30. Any inquiry concerning this communication or earlier communications from the examiner should be directed to AIMEE LI whose telephone number is (571)272-4169. The examiner can normally be reached on M-T 7:00am-4:30pm.

31. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

32. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Aimee J Li/
Primary Examiner, Art Unit 2183

24 October 2011